

EXHIBIT C

To: Edgars Nemše [edgars@stakehound.com]
Cc: Dean Tappen [dean.tappen@celsius.network]; Albert Castellana [albert@stakehound.com]; Harumi Urata-Thompson [harumi@celsius.network]; Connor Nolan [connor.nolan@celsius.network]
From: Neil Patel [neil.patel@celsius.network]
Sent: Sat 1/23/2021 7:00:32 PM (UTC)
Subject: Re: Please send stETH

23-01138-110 Doc 61-3 Filed 09/08/23 Entered 09/08/23 16:25:16 Exhibit C - January 23, 2021 Email Pg 2 of 10

Hi Edgars,
We confirm receipt of the additional 25,436.508489461 stETH.

Thanks,
Neil

On Sat, Jan 23, 2021 at 1:34 PM Edgars Nemše <edgars@stakehound.com> wrote:

Hi all,
We've just issued the remaining stETH to
you <https://etherscan.io/tx/0x5d2f4f3ac87b5809d07ccff64be02004196dd6aabeffa4489b7c5adf74b9b3bb>

Everything looks good on your end?

Best,
Edgars

On Sat, 23 Jan 2021 at 19:09, Edgars Nemše <edgars@stakehound.com> wrote:

Hi Neil,
Thanks for confirming!

We'll proceed with the issuance shortly.

Best,
Edgars

On Sat, 23 Jan 2021 at 18:59, Neil Patel <neil.patel@celsius.network> wrote:

Hi Edgars,
We successfully received the test transfer.

Thanks,
Neil

On Jan 23, 2021, at 11:52 AM, Edgars Nemše <edgars@stakehound.com> wrote:

Hi Dean,
My pleasure!

We're starting the issuance of stETH, and as a first step we've minted 0.01 stETH as a test to the address 0x38DE8739A69f31fDAAf596EcA87a31f282Bb625E as you requested <https://etherscan.io/tx/0x2358101da263ce6681e65d81ad961074b6a20868fe5aab0a8c929c1e9f95571f>

Could you please confirm that everything looks good from your end?

Best regards,

On Sat, 23 Jan 2021 at 00:08, Dean Tappen <dean.tappen@celsius.network> wrote:

Adding in a few more members of the trading team in case I am unable to assist this weekend.
Being a Packers fan I might be busy on Sunday!

On Fri, Jan 22, 2021, 5:05 PM Dean Tappen <dean.tappen@celsius.network> wrote:

Hey Albert,
Not a problem at all!

We will be around just give us a heads up and we can confirm the test transfer.

Regards,

Dean

On Fri, Jan 22, 2021, 5:02 PM Albert Castellana <albert@stakehound.com> wrote:

Dear Harumi and Dean - It's a pleasure meeting you.

I completely understand and appreciate the clarification. This is obviously a very important step for us so we want to do things right.

In order to help bring you up to speed with the situation, here is a pretty good overview of the process that we defined for the transfer:

<https://docs.google.com/document/d/1ixV5QYEHayqrm9d854eyGebwIj1uYvfoHbOOvhLzJs/edit>

Please, let me know if you have any questions about it.

Regarding the account - sorry I'm a bit slow today - I'm adding Edgars, co-founder and CTO, to the thread so that he can assist you in this part of the process.

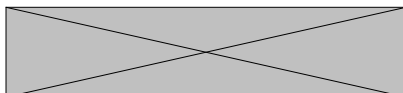
We are currently getting everything ready to go, but given that we are all around CET, we will probably have to finish this tomorrow/sunday.

Would that work for you?

Kind regards

Albert Castellana

CEO @ stakehound.com - Bringing Staking and DeFi together
+34 688989877



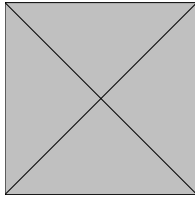
On Fri, Jan 22, 2021 at 10:51 PM Harumi Urata-Thompson <harumi@celsius.network> wrote:

Hello Albert,
Please send stETH to this address, starting with a test:

0x38DE8739A69f31fDAAf596EcA87a31f282Bb625E

Best regards,

--



Harumi Urata-Thompson

CFO CIO | Celsius

phone: 646-271-0003

StakeHound / Celsius

About stETH

StakeHound stETH is a liquid staking product for the Ethereum 2.0 blockchain.

Each stETH token represents service of staking an ETH token on StakeHound. StakeHound uses a rebasing ERC20 token to distribute staking rewards to all holders of stETH directly to their wallet.

The stETH token contract can be found at:

<https://etherscan.io/token/0xdf66b14d37c77f4e9b180ceb433d1b164f0281d>

Other projects listed by StakeHound include [stFIRO](#) and [stXEM](#).

StakeHound Technology

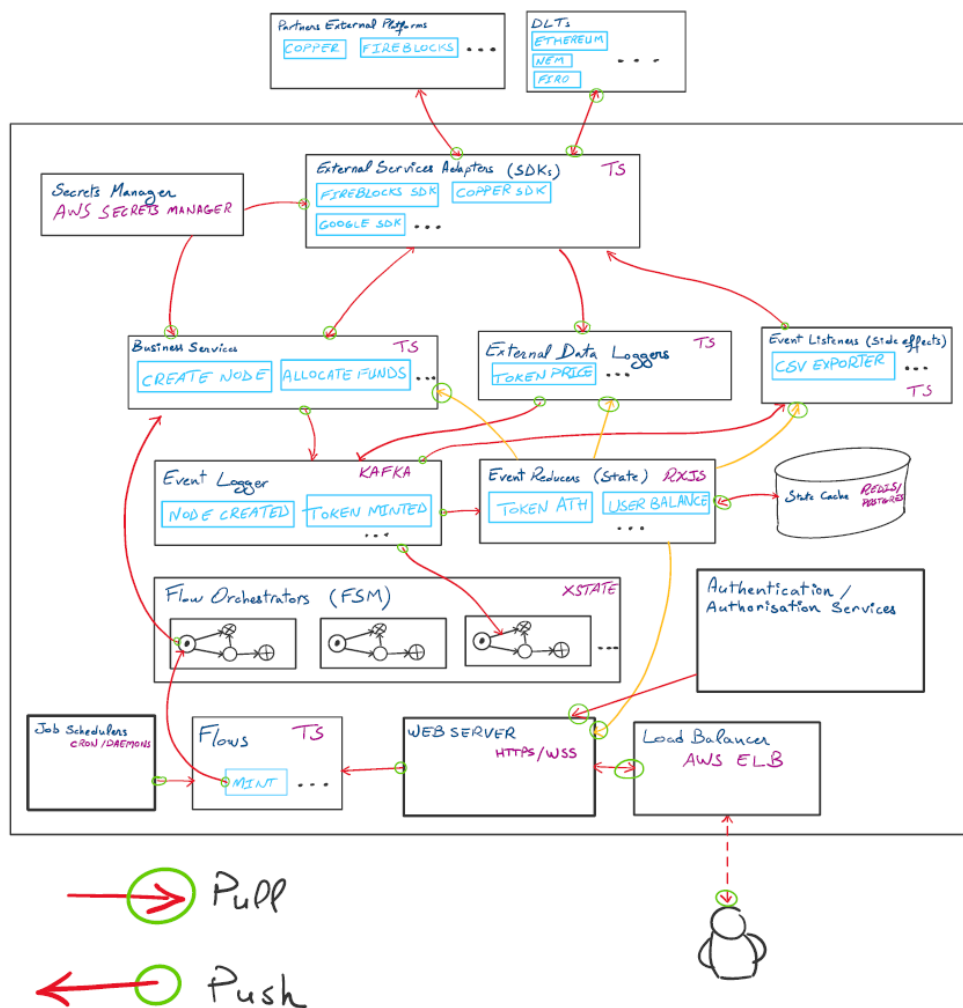
System Components

- Smart contract
- StakeHound backend
- Custody - Fireblocks
- Node Provider - Allnodes

StakeHound backend

The StakeHound backend is an event-sourced system built in TypeScript that ties together all external components of the system and performs actions such as minting, staking and distributing staking rewards. The Apache Kafka based event log acts as a source of truth for different modules of the system, as well as an audit log for all activity in the platform.

Here is a high level diagram of the architecture of the system that is being built:



Asset security

All assets held by StakeHound are held in institutional-grade custody, protected by MPC technology. Ethereum specifically is custodied in Fireblocks, and only allowed to be sent to the ETH2 deposit smart contract - all other actions require manual approval.

For withdrawal keys, StakeHound uses a sharded 3/4 BLS key solution provided by Fireblocks, where 2 keys are held by StakeHound and 2 by Fireblocks.

Backup of all Fireblocks custody keys is provided by Coincover.

Smart contract security

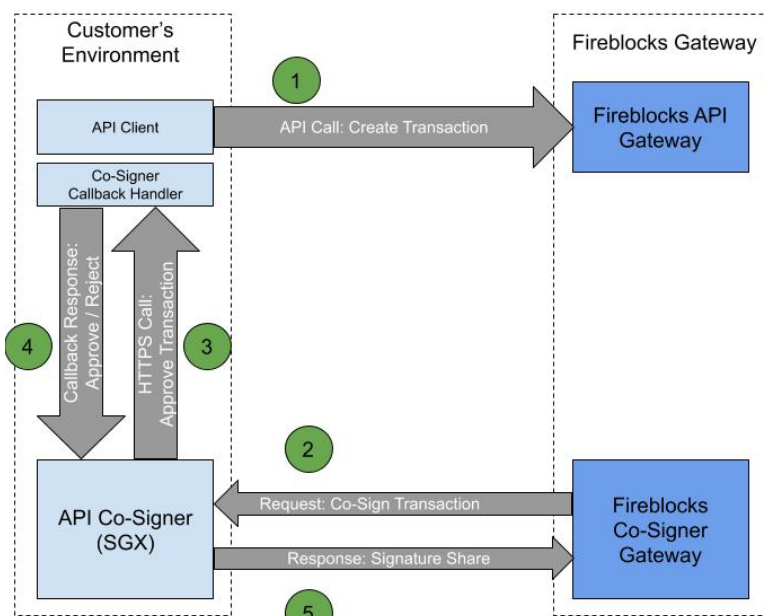
The smart contract source code is [open-source](#) and has been audited by [Quantstamp](#)

The StakeHound rebasing ERC20 smart contract has 2 roles - owner and supply controller.

The owner of the contract can set the supply controller, as well as pause the token and blacklist addresses. The owner is a Gnosis multi-sig consisting of hot and cold wallets of the StakeHound founders as well as external partners.

The supply controller role is in charge of minting, burning and issuing rewards. The supply controller is a Fireblocks vault protected by MPC technology. Currently all operations require manual approval from both founders Albert and Edgars.

In the near future it is planned to automate this process using the [Fireblocks Co-signer Callback feature](#). The callback will run a separate check to ensure no minting or rebasing operation can cause the total supply of the token to exceed the amount of tokens in StakeHound custody and whether the recipient address is found in our customer data sources. The co-signer runs on a separate cloud from the rest of the system.



Nodes

All StakeHound nodes are currently hosted by [Allnodes](#). Node keys are stored in AWS Secrets Manager. The node keys are only ever transferred encrypted with an Allnodes RSA public key.

Once live, there will be a public list of all addresses and nodes controlled by StakeHound.

Project 1 - Issuance of stETH for existing nodes

Transfer of existing Celsius nodes to StakeHound.

Motivation: Celsius has already set up a number of ETH2 nodes that they would like to transfer to StakeHound in order to exchange the underlying ETH for stETH.

Since the mnemonic that was used to create nodes has already been generated and can't be updated, it cannot be protected by the normal MPC guards. Therefore, the goal is to minimize the exposure of the mnemonic until such time that it is possible to change the withdrawal credentials of the nodes.

Parties:

- Celsius
- StakeHound
- Coincover

What needs to be transferred by Celsius:

- Mnemonic
- Deposit data and keystores with their password to import the nodes into StakeHound
- Access to the staking provider where the nodes are running - Staked.us
- Information about what address to deliver the stETH to

Transfer of Mnemonic

The goal of this process is to transfer the mnemonic in a way that is secure and verifiable, so that StakeHound can confidently own the assets to be able to issue the stETH tokens, without having to change the withdrawal key of the nodes until this capability is available.

In order to achieve this, we propose the use of a trusted third party - Coincover. Given that Coincover can never know the mnemonic, the proposed process looks as following:

1. Coincover will generate a RSA key-pair (Public and Private key).
2. Coincover will hold the private key in their secure vault.
3. Coincover will provide StakeHound with the Public key.
4. The Public Key will be passed on to Celsius and used to encrypt the mnemonic.
5. The encrypted mnemonic will then be passed on to StakeHound.

Never to Coincover.

6. When the time comes, StakeHound will approach Coincover and request the private key to decrypt the mnemonic and take full control of the funds.

The RSA encrypted mnemonic will be stored safely in multiple locations.

Transfer of other data

Keystores can be sent in plaintext, since they are encrypted and password protected.

The keystore password as well as access details for staked.us and the Celsius address for stETH will need to be shared using a secure channel. StakeHound will provide Celsius with a public RSA key that can be used to encrypt and send the data to StakeHound.

Issuing of stETH

When all of the required keys and data are transferred to StakeHound, StakeHound will mint an amount of stETH that matches the current balance of the transferred nodes to the address specified by Celsius.

Transition to MPC

As soon as ETH2 Phase 1.5 goes live, which will enable access to the funds locked in validators, StakeHound will work with Coincover to change the withdrawal credentials of the nodes to secure sharded BLS keys.

To ensure that the mnemonic is useless after the transition, StakeHound will create a new set of node keys as well as withdrawal keys.

The process will happen fully off-line and in person:

1. StakeHound employee visits Coincover on-site
2. StakeHound employee provides Coincover with the encrypted mnemonic data
3. Coincover decrypts the data on their offline used to store the RSA key
4. Coincover transfers the mnemonic through an offline medium to a clean off-line machine provided by StakeHound
5. StakeHound employee uses this machine to generate and sign the required set of transactions
6. The transactions are transferred out of the offline machine to a different machine for broadcasting to the public ledger
7. The transactions are broadcast to the network
8. The mnemonic is now useless

Project 2 - API for purchasing of stETH

StakeHound will work with Celsius to produce an API that allows Celsius to acquire stETH in exchange for ETH in an automated fashion.

On a high level, the process is:

- Celsius sends StakeHound ETH from and to a specific account
- StakeHound sends Celsius the same amount in stETH, to a specific account

Sending ETH

Given that Celsius has already fully gone through our KYC/AML process:

1. StakeHound will issue Celsius a single address to send ETH to during the setup phase of the integration.
2. Celsius will need to specify a list of addresses to whitelist, from which they will send ETH. These addresses must have been whitelisted before sending any ETH.

In the future, if new addresses need to be whitelisted, Celsius will send a request to StakeHound to do so to edgars@stakehound.com as well as pascal@altcoinomy.com

If Celsius has a Fireblocks Vault, we can use the Fireblocks Network to do 0-confirmation transactions and abstract away addresses.

Outputting stETH

For receiving stETH, there are a few options:

- Send all stETH to a single address specified by Celsius in the setup phase
- Send stETH to a mapping of addresses specified in the setup phase:
 - ETH from Address_A -> stETH to Address_C
 - ETH from Address_B -> stETH to Address_D
- Send stETH to the same address ETH was received from (This can only work for ETH not other stakedTokens)
 - ETH from Address_A -> stETH to Address_A

Finally, if it is required, we can build an API to set the output address and amount per transaction, from a list of pre-whitelisted addresses.

Other tokens

This document talks about stETH, but the same mechanism can be extended to other tokens with minimal adaptations in the future. Other tokens will also include the possibility to go from stTokens to native tokens.